Statistica Sinica Preprint No: SS-2016-0397.R1	
Title	Network Inference From Grouped Observations Using Hub Models
Manuscript ID	SS-2016-0397.R1
URL	http://www.stat.sinica.edu.tw/statistica/
DOI	10.5705/ss.202016.0397
Complete List of Authors	Yunpeng Zhao and Charles Weko
Corresponding Author	Yunpeng Zhao
E-mail	yzhao15@gmu.edu
Notice: Accepted version subject to English editing.	

Network Inference From Grouped Observations Using Hub Models

YUNPENG ZHAO[1] AND CHARLES WEKO[2]

George Mason University[1] *and United States Army*[2]

Abstract: In medical research, economics, and the social sciences data frequently appear as subsets of a set of objects. Over the past century a number of descriptive statistics have been developed to infer network structure from such data. However, these measures lack a generating mechanism that links the inferred network structure to the observed groups. To address this issue, we propose a model-based approach called the *Hub Model* which assumes that every observed group has a leader and that the leader has brought together the other members of the group. The performance of Hub Models is demonstrated by simulation studies. We apply this model to the characters in a famous 18^{th} century Chinese novel.

Key words and phrases: Social network analysis, affiliation network, expectation-maximization algorithm, half weight index, Dream of the Red Chamber.

1 INTRODUCTION

A network can be denoted by $N = (V, E)$, where $V = \{v_1, v_2, ..., v_n\}$ is the set of n nodes, and E is the set of edges between nodes. In this article, we focus on symmetric weighted networks represented by an $n \times n$ adjacency matrix, A, where the element A_{ij} measures the relationship strength between nodes v_i and v_j.

Traditionally, statistical network analysis focuses on modeling *observed* network structure (e.g., highway systems or electrical transmission grids). In this situation, nodes are well defined and the physical links between nodes is observable (Hiller and Lieberman, 2001; Newman, 2011). However, in some fields of research (e.g., the social sciences) network structure is not explicit. In these fields, the observable data are groups of individuals and a model is presumed to produce the groups. The fundamental task is to estimate model parameters from such data.

Wasserman and Faust (1994) introduce inference of relationships with the example of children attending birthday parties. In their example, the children act as nodes in the network and the birthday parties represent subsets of children.

In this paper, a collection of nodes observed in the same sample is called a *group* and a dataset is called *grouped data*. In Wasserman and Faust's example, each party defines a group and the set of all parties is the grouped data. Two individuals are said to *co-occur* if they appear in the same group.

One common technique used to estimate an adjacency matrix from grouped data is to count the number of times that a pair of nodes appears in the same group (Zachary, 1977; Freeman et al., 1989; Wasserman and Faust, 1994; Kolaczyk, 2009; Brent et al., 2011). Frequently, a threshold is applied to this count to create an unweighted adjacency matrix; however, Choudhury et al. (2010) show that the characteristics of networks inferred by this technique are sensitive to the threshold. We adopt a generalized version of the inter-citation frequency (Kolaczyk, 2009) which measures the number of times a pair of nodes is observed to co-occur in the dataset. We refer to this measure as the *co-occurrence matrix*.

An alternative technique, called the *half weight index* (Cairns and Schwager, 1987), estimates an adjacency matrix by the frequency that two nodes co-occur given that one of them is observed. This addresses a shortcoming of the co-occurrence matrix in which nodes that appear rarely can be estimated to have a weak relationship even though the relationship is quite strong (Voelkl et al., 2011).

The co-occurrence matrix and half weight index both have probabilistic interpretations. The co-occurrence matrix estimates the probability that two nodes will be observed together. The half weight index estimates the probability that two nodes will be observed together given that one of them is observed. However, these are not equivalent to the probability of an active relationship between nodes. In fact, neither of these techniques describe the process which leads to the generation of the observed groups. It is unclear how these descriptive statistics relate to the grouped data in these methods.

We propose a model-based approach for grouped data generation which we refer to as the *Hub Model* because each observed group is assumed to be brought together by a hub node (see Figure 1).

The Hub Model is fundamentally different from classical network models such as the stochastic blockmodel and its variants (Holland et al., 1983; Airoldi et al., 2008), the exponential random graph models (Frank and Strauss, 1986; Robins et al., 2007), the latent space model and its variants (Hoff et al., 2002; Handcock et al., 2007), among others (see Goldenberg et al. (2010) for a comprehensive review). These models focus on modeling the statistical behavior of the network, that is, they treat the network as the observed data. By contrast, the Hub Model treats the network as latent governing the grouping behavior of a population. Our task is to estimate the latent network (i.e., the adjacency matrix) from the observed group data. In this article,

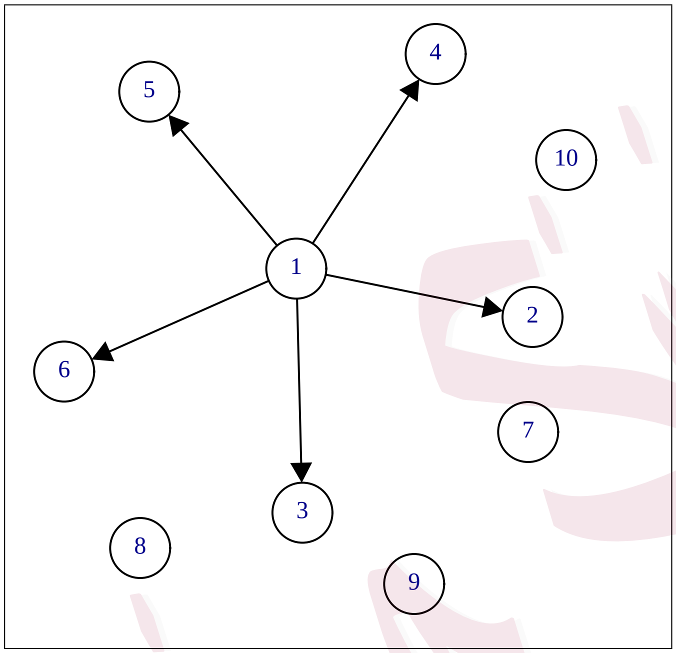


Figure 1: The generating mechanism of the Hub Model is demonstrated on a group of 10 nodes. In the observed sample, nodes $v_1, \ldots, v_6$ are members of the group while nodes $v_7, \ldots, v_{10}$ are not members of the group. The observed group is the result of the hub node, v_1, bringing together nodes $v_2, \ldots, v_6$.

we treat the adjacency matrix as fixed parameters and make no structural assumption about it. If there were *a priori* information about the latent network, such as that it follows the stochastic blockmodel or the exponential random graph model, then one could take a Bayesian approach and use this model as *a prior*. For more discussion, refer to Section 7.

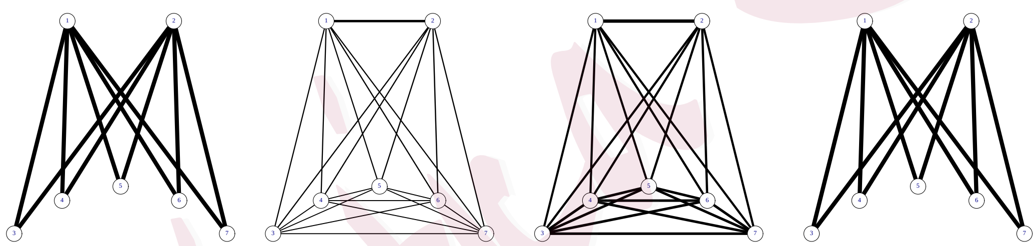
The Hub Model belongs to the family of finite mixture models which has been applied in many different situations including text classification

(Carreira-Perpinan and Renals, 2000), topic models (Anandkumar et al., 2015), fingerprint identification (Vretos et al., 2012), and product recommendation (Colace et al., 2015).

Hub Models have the advantage that relationship strength is both mathematically well defined and practical to researchers. In the Hub Model, A_{ij}, is defined as the probability that node v_i will include node v_j when v_i is the hub node of a group. The formal definition of the Hub Model will be given in Section 3.

As an introduction to Hub Models, consider the hypothetical relationships in Figure 2a. In this example there is a pair of nodes, v_1 and v_2, which never directly pair to each other; however, they have an 80% chance of interacting with five nodes. That is, $A_{ij} = 0.8$ for all $i \leq 2$ and $j \geq 3$ while $A_{ij} = 0$ otherwise. In Figure 2b, the co-occurrence matrix mistakenly assigns a relatively strong relationship to nodes v_1 and v_2 because they often co-occur. In Figure 2c, the half weight index arrives at a similar conclusion. In both Figures 2b and 2c, the non-existent relationship between nodes v_1 and v_2 is estimated to be stronger than all other relationships. By contrast, the Hub Model in Figure 2d clearly captures the relationships of the population.

To the best of our knowledge, there have been limited attempts to apply model-based approaches to these data. Rabbat et al. (2008) provide an application for telecommunication networks. They modeled group formation as a random walk from a source node to a terminal node. This model assumed a distinctly different process of group formation from Hub Models. The nodes along the path were subjected to an unknown permutation to account for the lack of order information. Treating permutations as missing data, Rabbat et al. employed a *Monte Carlo EM* algorithm based on importance sampling to estimate the parameters of the model.



(a) Truth (b) Co-occurrence (c) Half Wt. Index (d) Hub Model

Figure 2: Comparison of Estimation Techniques

In the following sections we present a formal description of the grouped data structure, review existing techniques, and define Hub Models. Then we address Hub Model identifiability and provide a theorem that proves that a symmetric adjacency matrix is a sufficient condition for identifiability. We propose an EM algorithm to solve the maximum likelihood estimator of the Hub Model. We then evaluate the model performance by simulation studies. We apply the Hub Model to infer the relationships among the characters of the 18^{th} century Chinese novel, *Dream of the Red Chamber*. We close with a discussion of how the size of the population impacts model efficiency and ways to incorporate network structure assumptions to simplify the model.

2 GROUPED DATA

2.1 Data Structure

For a population of n individuals, $V = \{v_1, \ldots, v_n\}$, we observe T subsets of the global population, $\{V^{(t)}|V^{(t)} \subseteq V, t = 1, ..., T\}$. Each observed subset can be coded as an n length row vector $G^{(t)}$ where:

$$G_i^{(t)} = \begin{cases} 1 & \text{if } v_i \in V^{(t)} \\ 0 & \text{if } v_i \notin V^{(t)} \end{cases}$$

The full set of observations is denoted by a $T \times n$ matrix, G. The t^{th} row of G is $G^{(t)}$.

2.2 Existing Methods

Inferring relationships from grouped data relies on descriptive statistics which count the number of times that two nodes are observed together. We focus on two popular techniques which estimate probabilities of individual behavior.

A simple measure of grouped data is the *co-occurrence matrix*. Versions of this technique appear throughout the literature under many names and notations including: *capacity matrix* (Zachary, 1977), *sociomatrix* (Wasserman and Faust, 1994), *inter-citation frequency* (Kolaczyk, 2009), *cocitation matrix* (Newman, 2011), and *strength* (Brent et al., 2011).

A co-occurrence matrix, O, is an $n \times n$ symmetric matrix, defined by:

$$O = \frac{G'G}{T}, \tag{2.1}$$

which estimates the frequency that the nodes v_i and v_j are observed in the same group.

One shortcoming of the co-occurrence matrix is that it estimates the probability that two nodes *will be observed* to co-occur in a given observation. That is, if two nodes have a strong relationship, but appear in the dataset infrequently, the co-occurrence matrix will estimate a low probability that the two nodes *will be observed* to co-occur.

As an example, consider four nodes $v_1, \ldots, v_4$ and the grouped data represented in Table 1. For this dataset, both $O_{1,2} = \frac{2}{5}$ and $O_{3,4} = \frac{2}{5}$. However, notice that every time node v_3 is present node v_4 is also present. A researcher

	Node			
Event	v_1	v_2	v_3	v_4
1	1	0	0	0
2	1	1	0	0
3	1	1	0	0
4	1	0	1	1
5	0	1	1	1

Table 1: Notional Grouped Data

may conclude that there is some aspect of the relationship between nodes v_3 and v_4 which has been understated.

As an alternative, the *half weight index* estimates the probability that two nodes will be observed to co-occur given that one of them is observed (Cairns and Schwager, 1987).

The half weight index has been introduced in a number of equivalent forms (Dice, 1945). Computationally, the most direct form is:

$$H_{ij} = \frac{2\sum_t G_i^{(t)} G_j^{(t)}}{\sum_t G_i^{(t)} + \sum_t G_j^{(t)}}. \tag{2.2}$$

Returning to the example in Table 1, we can see that $H_{1,2} = \frac{4}{7}$ while $H_{3,4} = \frac{4}{4}$. Therefore, the half weight index infers a different network than the co-occurrence matrix.

3 HUB MODELS

3.1 Generating Mechanism

Hub Models (HM) assume that each group is a star subgraph on the global population. The hub node connecting the observed group is represented by an n length row vector, $S^{(t)}$, where

$$S_i^{(t)} = \begin{cases} 1 & \text{if } v_i \text{ the hub node of sample } t, \\ 0 & \text{otherwise.} \end{cases}$$

There is one and only one element of $S^{(t)}$ that is equal to 1.

Each group is independently generated by a two step process.

1. The hub node is drawn from a multinomial distribution with parameter $\rho = (\rho_1, ..., \rho_n)$, i.e., $\rho_i = \mathbb{P}(S_i^{(t)} = 1)$. The following constraint applies:

$$\sum_i \rho_i = 1. \tag{3.3}$$

2. The hub node, v_i, chooses to include v_j in the group with probability A_{ij}, i.e., $A_{ij} = \mathbb{P}(G_j^{(t)} = 1 | S_i^{(t)} = 1)$.

In most practical applications, the hub node of each group is unknown. This article focuses on this case. We refer to the model where leaders are known as the Known Hub Model (KHM).

Since the co-occurrence matrix and half weight index produce a symmetric adjacency matrix, we assume the Hub Model adjacency matrix is symmetric. The symmetry condition will be shown to ensure the identifiability of the Hub Model when group leaders are unobserved (Supplemental Material S1.2).

Further, we assume that the hub node will always include itself in the group, i.e. $A_{ii} = 1$ for all i.

This generating mechanism implies that each observed group is independent of every other observed group. In particular, $G^{(t)}$ is not a transformation of $G^{(t-1)}$ and the order in which groups are observed contains no information about the relationships between group members. Researchers often collect data in such a way to ensure this property (Bejder et al., 1998).

3.2 Likelihood of the Hub Model

Under the HM, the probability of an observation has the form of a finite mixture model with n components:

$$\mathbb{P}(G^{(t)}|A, \rho) = \sum_{i=1}^{n} \rho_i G_i^{(t)} \prod_j A_{ij}^{G_j^{(t)}} (1 - A_{ij})^{1-G_j^{(t)}}. \tag{3.4}$$

By taking the log of the product of individual observed groups, the log likelihood function for the full set of observations is:

$$\mathcal{L}(G|A,\rho) = \sum_t \log\Big[\sum_{i=1}^n \rho_i G_i^{(t)} \prod_j A_{ij}^{G_j^{(t)}} (1-A_{ij})^{1-G_j^{(t)}}\Big]. \tag{3.5}$$

Solving the MLE of HM is an optimization problem with the equality constraints $\sum_i \rho_i = 1$, and $A_{ij} = A_{ji}$ for all i and j. From (3.5), we denote the log likelihood function as $\mathcal{L}(G|A,\rho)$. This gives the following Lagrange function:

$$\Lambda(G|A,\rho) = \mathcal{L}(G|A,\rho) - \lambda_o[(\sum_i \rho_i) - 1] - \sum_{i<j} \lambda_{ij}(A_{ij} - A_{ji}). \tag{3.6}$$

The log likelihood does not have a closed-form solution for the MLE. Instead we will derive estimating equations which can be incorporated into an Expectation Maximization algorithm. Before doing so we investigate the identifiability of the Hub Model.

A basic requirement for any model is *identifiability*. For Hub Models, this means for any two sets of parameters $\{A,\rho\}$ and $\{A^*,\rho^*\}$:

$$\mathbb{P}(G = g|A,\rho) = \mathbb{P}(G = g|A^*,\rho^*) \ \forall g \implies A = A^*, \rho = \rho^*. \tag{3.7}$$

The generating mechanism for Hub Models is equivalent to a finite mixture model of multivariate Bernoulli random variables. In general, such a model is not identifiable (Teicher, 1961). This shortcoming does not prevent such models from being useful in many applications. For example, when dealing with classification problems where the researcher only has to identify which component density an observation came from, this type of mixture can be effectively used (Carreira-Perpinan and Renals, 2000). In such a situation, the individual parameters of the multivariate Bernoulli random variables are not of interest. However, the issue of identifiability presents a challenge in our application because we are specifically interested in the individual parameters of the adjacency matrix.

If no constraint is put on the adjacency matrix, the model is unidentifiable. The following theorem establishes a sufficient condition for identifiability. See Supplemental Material S1 for more details.

Theorem 1. Let A and A^* be symmetric adjacency matrices with $A_{ii} = A^*_{ii} = 1$ for all i, $A_{ij} < 1$ and $A^*_{ij} < 1$ for all $i \neq j$. If $\mathbb{P}(g|A, \rho) = \mathbb{P}(g|A^*, \rho^*)$ for all g, then $\{A, \rho\} = \{A^*, \rho^*\}$.

It is worth noticing that even though symmetry of the adjacency matrix is a natural assumption, it is only a sufficient condition for identifiability according to Theorem 1. For future work, we will explore other assumptions to ensure identifiability and ultimately find a necessary and sufficient condition.

3.3 Estimating Equations

In Supplemental Materials S2, we derive (3.8) and (3.9) which are estimating equations that the MLE must satisfy. The maximum likelihood estimator of HM does not have a closed-form solution for the parameters because the right hand side of the estimating equations includes the estimated parameters. Next we will show that solving these equations iteratively is equivalent to an EM algorithm. The details of the EM algorithm will be given in the next section.

$$\hat{A}_{xy} = \frac{\sum_t G_y^{(t)} \mathbb{P}(S_x = 1|G^{(t)}) + \sum_t G_x^{(t)} \mathbb{P}(S_y = 1|G^{(t)})}{\sum_t \left[\mathbb{P}(S_x = 1|G^{(t)}) + \mathbb{P}(S_y = 1|G^{(t)})\right]}. \tag{3.8}$$

$$\hat{\rho}_x = \frac{\sum_{t=1}^{T} \mathbb{P}(S_x^{(t)} = 1|G^{(t)})}{T}. \tag{3.9}$$

4 EM ALGORITHM

The estimating equations shown above depend on the probability $\mathbb{P}(S_x^{(t)} = 1|G^{(t)})$. This implies an algorithm updating $\{\hat{A}, \hat{\rho}\}$ and $\mathbb{P}(S_x^{(t)} = 1|G^{(t)})$ iteratively, which can be fitted into the general framework of an EM algorithm.

The key technique of any EM algorithm is to formulate a complete data model then solve the model as if some data is observed and other data is missing. In this case, the Known Hub Model serves as the complete data model, G is the observed data, and S is the missing data. Each iteration of the EM algorithm consists of an expectation step followed by a maximization step (McLachlan and Krishnan, 2008).

E-Step

Since the log likelihood function of the complete data model is linear in the unobserved data, the E-Step (on the $(m+1)^{th}$ iteration) simply requires calculating the current conditional expectation of $S_i^{(t)}$ given the observed data (see McLachlan and Krishnan (2008) for detailed explanation).

$$\begin{aligned} E[S_x^{(t)}|G^{(t)}] &= \mathbb{P}(S_x^{(t)}=1|G^{(t)}) \\ &= \frac{\rho_x G_x^{(t)} \prod_j A_{xj}^{G_j^{(t)}} (1-A_{xj})^{1-G_j^{(t)}}}{\sum_{i=1}^n \rho_i G_i^{(t)} \prod_j A_{ij}^{G_j^{(t)}} (1-A_{ij})^{1-G_j^{(t)}}} \end{aligned} \tag{4.10}$$

M-Step

The M-Step replaces $\mathbb{P}(S_x^{(t)}=1|G^{(t)})$ on the right hand side of (3.8) and

(3.9) with $E[S_x^{(t)}|G^{(t)}]$ from (4.10).

Algorithm

Algorithm 1 illustrates the details of the Hub Model.

Several standard techniques are used to improve the performance of the EM algorithm. Firstly, we run the EM algorithm 10 times with different staring points and choose the solution with the highest likelihood. Secondly, we limit the number of iterations applied to a starting point. This second treatment is based in part on the observation that when this algorithm has a bad starting point, it will take a very long time to converge and the point that it converges to is not close to the maximum. As a final step, we treat any $\hat{A}_{xy} \leq 10^{-4}$ as $\hat{A}_{xy} = 0$. We apply this finishing step to remove clutter from the returned solutions.

5 SIMULATION

In order to perform simulations, we generate parameters $\{A, \rho\}$ using the following techniques.

For ρ, we select n random numbers, X_i, uniformly and divide each random

---

Data: G
Result: $\hat{A}, \hat{\rho}$
Initialize:
$\mathcal{L}(G|\hat{A}) = -\infty$
for rep=1 to 10 **do**
    Initialize:
    $\hat{A}_{ij}^{(0)} = unif(0,1)\ \forall\{i,j\}$
    $X_i = unif(0,1)\ \forall i$
    $\hat{\rho}_i^{(0)} = \frac{X_i}{\sum_k X_k}$
    $\Delta\mathcal{L}(G|A^{(0)}) = 10^4$
    counter=1
    while $|\frac{\Delta\mathcal{L}(G|A^{(m+1)})}{\mathcal{L}(G|A^{(m)})}| > 10^{-4}$ and counter < 100 **do**
        E-Step
          Update $\mathbb{P}(S_k^{(t)} = 1|G^{(t)})$ by Equation 4.10
        M-Step
          Update $A^{(m+1)}$ by Equation S2.10
          Update $\rho^{(m+1)}$ by Equation S2.13
        $\Delta\mathcal{L}(G|A^{(m+1)}) = \mathcal{L}(G|A^{(m+1)}) - \mathcal{L}(G|A^{(m)})$
        counter=counter+1
    end
    if $\mathcal{L}(G|A^{(m+1)}) > \mathcal{L}(G|\hat{A})$ **then**
        if $\hat{A}_{ij} \leq 10^{-4}$ **then**
            $\hat{A}_{ij} = 0$
        else
            $\hat{A}_{ij} = A_{ij}^{(m+1)}$
        end
    end
end

Algorithm 1: Expectation Maximization Algorithm for the Hub Model

number by the sum of all X_i's. That is, $\rho_i = \frac{X_i}{\sum_i X_i}$.

We use a two step process to generate the adjacency matrix. First, we create a symmetric unweighted undirected random graph on n nodes using the configuration model (Jackson, 2010) with a power law degree distribution $\mathbb{P}(k) \propto k^{-\eta}$, where k is the possible value of the node degree. We assume a power law degree distribution because it is commonly believed that many real world social networks have this property Newman (2011). In all simulation studies, we choose $\eta = 2$, because many networks are reported to have power between 2 and 3 and a power of 2 generates the most dense networks, which is a more challenging setup. We refer to this unweighted graph as the *structure* of the network.

Each edge in the graph is then assigned a relationship strength with a beta distribution,

$$A_{ij} = \begin{cases} Beta(\alpha, \beta) & \text{if there is an edge between } v_i \text{ and } v_j \\ 0 & \text{otherwise} \end{cases}$$

We simply let $A_{ji} = A_{ij}$ to ensure symmetry. We set $\alpha = 1$ and $\beta = 4$ in the beta distribution so that the average relationship strength is less than 0.5, which we believe is realistic.

In Tables 2 and 3, we consider five different network sizes $n = 10, 20, 50, 100, 150$. For the first two cases, we set the minimum node degree to be 1 in the power law distribution. And for the last three cases, we set the minimum degree to be 5 in order to make sure the networks are not too sparse. For each size, we generate 100 sets of parameters (A, ρ) using the setup described above. For each (A, ρ), we generate a dataset with T groups. Each average and standard deviation are calculated over this 100 datasets. We use 9 different values of $T = 100, 200, 500, 1000, 2000, 5000, 10000, 20000, 50000$.

We first measure the ability of the estimated adjacency matrix $\hat{A}$ to correctly identify the structure. To do this we define true positives and true negatives as follows:

$$TP = \sum_{i<j} \mathbb{1}_{(A_{ij}>0)} \mathbb{1}_{(\hat{A}_{ij}>10^{-4})},$$

$$TN = \sum_{i<j} \mathbb{1}_{(A_{ij}=0)} \mathbb{1}_{(\hat{A}_{ij}\leq 10^{-4})}.$$

Here, v_i and v_j are considered to have no relationship if the estimated link strength is below 10^{-4}. False positives and false negatives are calculated similarly. We use the Matthews correlation coefficient (MCC) to measure the identification of the structure because it is a binary classification measure that accounts for situations where the number of ones is significantly different than the number of zeros (Liu et al., 2015). Based on our setup, our simulated structures will have many more zeros than ones.

For the non-zero elements A_{ij}, we further evaluate the difference between the numerical values of A_{ij} and $\hat{A}_{ij}$ by calculating the mean absolute error (MAE) of non-zero A_{ij},

$$MAE(A) = \frac{\sum_{i<j} |\hat{A}_{ij} - A_{ij}| \mathbb{1}_{(A_{ij}>0)}}{\sum_{i<j} \mathbb{1}_{(A_{ij}>0)}}.$$

We also report the average run time and the average number of iterations for the EM algorithm when the simulation is run on an Intel Pentium CPU G2030 at 3.00 GHz with 4.00GB of RAM.

The first observation from Tables 2 and 3 is that for a fixed value of n the average error of both the MCC and the MAE decline as the number of observations increases. By contrast, for a fixed number of observations, the average error increases as the number of nodes increases.

The standard deviation of estimates generally improves once the number of observations exceeds the number of parameters in the model. For example, with 100 nodes there are roughly 10,000 parameters to estimate, thus samples of only 2,000 or 5,000 observations demonstrate high standard deviations.

Finally, average run time generally increases as the number of observations increases and the number of nodes increases. An important factor affecting the run time is the number of iterations the EM algorithm performs before converging. In Table 2 the number of iterations declines as observations increase until it appears to approach a minimum number of iterations. Table 3 provides further insight as the number of iterations generally increases until the number of observations is roughly equal to the number of parameters in the model after which the iterations declines. Up to that point, the algorithm quickly converges to an adjacency matrix which is sparser than the true adjacency matrix due to the insufficient sample size. The implication of these declining iterations is that run time is not strictly a function of the size of the dataset, but the relationship between the number of nodes and the number of observations.

	$n = 10$					
Obs	Avg MCC	StDev MCC	Avg MAE(A)	StDev MAE(A)	Avg Run Time (sec)	Avg Iterations
100	0.8010	0.0977	0.0533	0.0219	0.0472	20.258
200	0.8929	0.0903	0.0349	0.0128	0.0431	16.670
500	0.9487	0.0530	0.0212	0.0071	0.0411	13.618
1000	0.9770	0.0364	0.0147	0.0047	0.0369	12.011
2000	0.9865	0.0279	0.0102	0.0030	0.0353	10.613
5000	0.9984	0.0115	0.0067	0.0019	0.0298	9.604
10000	0.9988	0.0086	0.0045	0.0014	0.0295	9.416
20000	0.9994	0.0060	0.0035	0.0009	0.0305	9.327
50000	1	0	0.0020	0.0006	0.0316	9.210
	$n = 20$					
100	0.6727	0.0972	0.0833	0.0210	0.1005	21.007
200	0.7984	0.0756	0.0599	0.0154	0.0992	19.961
500	0.8781	0.0576	0.0340	0.0079	0.1039	17.793
1000	0.9147	0.0594	0.0225	0.0056	0.1131	15.418
2000	0.9360	0.0612	0.0150	0.0033	0.1473	13.803
5000	0.9734	0.0367	0.0099	0.0024	0.1653	11.571
10000	0.9842	0.0393	0.0069	0.0019	0.1806	10.662
20000	0.9937	0.0187	0.0048	0.0013	0.2052	10.260
50000	0.9989	0.0070	0.0031	0.0006	0.2320	9.888

Table 2: Average and Standard Deviation of Mean Absolute Error as Observations Increase

Obs	Avg MCC	StDev MCC	Avg MAE(A)	StDev MAE(A)	Avg Run Time (sec)	Avg Iterations
	$n = 50$					
100	0.3454	0.0503	0.1680	0.0139	0.2272	5.261
200	0.3987	0.0622	0.1368	0.0081	0.9216	16.237
500	0.5815	0.0668	0.0936	0.0085	2.7233	36.148
1000	0.8499	0.0302	0.0526	0.0049	2.6903	38.222
2000	0.9013	0.0176	0.0345	0.0030	2.3761	24.713
5000	0.9127	0.0193	0.0212	0.0017	2.8953	17.802
10000	0.9074	0.0259	0.0145	0.0012	5.1788	15.343
20000	0.9080	0.0327	0.0104	0.0008	7.1548	13.932
50000	0.9142	0.0383	0.0065	0.0006	12.190	12.866
	$n = 100$					
100	0.2620	0.0352	0.1955	0.0096	0.2058	2.040
200	0.3187	0.0346	0.1756	0.0109	0.2922	2.533
500	0.3495	0.0519	0.1359	0.0070	1.8683	9.151
1000	0.3857	0.0498	0.1109	0.0074	6.9431	25.852
2000	0.5343	0.1055	0.0748	0.0100	14.6644	44.035
5000	0.8236	0.1469	0.0351	0.0080	17.5031	34.544
10000	0.9128	0.0826	0.0219	0.0028	19.4031	23.370
20000	0.9355	0.0579	0.0148	0.0015	22.4366	17.494
50000	0.9484	0.0282	0.0092	0.0006	33.8123	13.905
	$n = 150$					
100	0.2247	0.0366	0.1994	0.0105	0.3373	1.536
200	0.2674	0.0316	0.1909	0.0081	0.3705	1.547
500	0.2965	0.0431	0.1632	0.0091	0.8822	2.623
1000	0.2625	0.0600	0.1363	0.0067	7.4969	11.65
2000	0.2354	0.0628	0.1247	0.0089	42.4597	47.525
5000	0.2700	0.1402	0.1075	0.0144	98.8080	75.973
10000	0.4276	0.2247	0.0822	0.0252	150.6061	72.416
20000	0.6025	0.2601	0.0532	0.0280	184.3534	60.144
50000	0.7602	0.2441	0.0275	0.0230	217.9005	41.975

Table 3: Average and Standard Deviation of Mean Absolute Error as Observations Increase (continued)

6 DATA ANALYSIS

In this section, we perform data analysis on the 18^{th} century Chinese novel, *Dream of the Red Chamber*. The observed groups in this dataset do not necessarily conform to the Hub Model assumption. However, we will show that even without this assumption being explicitly valid, important information about the relationships can be estimated.

The Supplemental Materials S3 include two additional data sets estimating co-sponsorship of legislation in the Senate of the 110^{th} United States Congress and the dispersion of plant species across North America.

As noted by Kolaczyk (2009), a significant challenge with estimating the parameters of implicit networks is that for a real world dataset there is usually no way to verify the extent to which the estimate matches reality. That is, there is no so-called "ground truth" or "golden standard" to compare the estimated results against. Therefore, it is useful to analyze data about which there is some qualitative knowledge of the relationships between nodes. To this end, we construct a dataset of characters from *Dream of the Red Chamber*. Since novels contain a qualitative social structure that is familiar to readers, the results of quantitative analysis can be compared to this standard.

This novel is chosen for two reasons. Firstly, the relationships between the characters are subtle and complex. Secondly, the novel has been carefully studied by scholars. Therefore, the story presents a challenge to the task of estimating relationships and there is a body of knowledge to compare the estimates against.

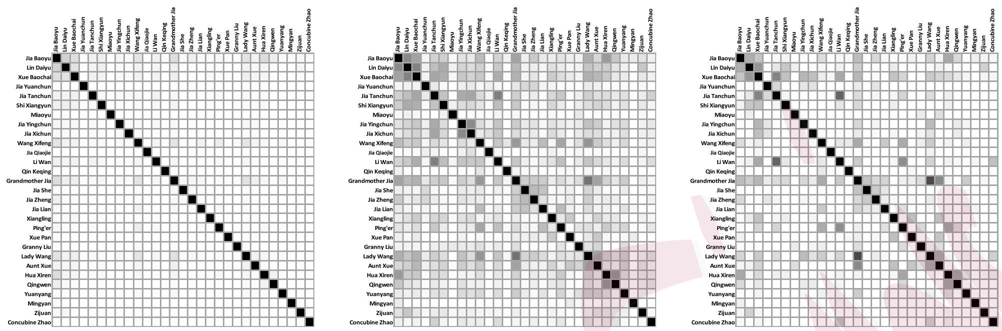
Traditionally datasets are built from novels by carefully reading the text and identifying dyadic interactions between characters based on criteria established by the researchers, e.g., characters A and B have a conversation (MacCarron and Kenna, 2013). This method may construct high quality datasets; however, in order to identify interactions, it requires readers who can read the novel and have time to build the datasets. Since *Dream of the Red Chamber* is written in classical Chinese and the English translation runs over 2,600 pages, directly generating the dataset would be excessively time consuming.

Therefore, we built the dataset using text mining and define a group as characters who co-occur in the same paragraph. Paragraphs with no characters named in them are ignored. For a complete description of the text mining protocol, see Supplemental Materials S5.

We analyze the relationships of 29 important characters. The character names presented here are based on the original pinyin pronunciations and the David Hawkes translation (Hawkes, 1974). A Chinese version of the novel was used for text-mining. The complete novel contains 120 chapters, but we focus on the first 80 because it is commonly believed that the last 40 chapters are written by a different author and may not reflect the original themes of the novel (Hsueh-Chin, 2016). The resulting dataset has 1,389 observations of groups containing at least one of the 29 characters.

In Figure 3, the adjacency matrix is represented as an $n \times n$ grid where the $i^{th} \times j^{th}$ cell represents the relationship between nodes v_i and v_j. The relationship strength is represented by the cell's color. Nodes with weak relationships have light cells while nodes with strong relationships have dark cells. Cells representing relationships of intermediate strength are shaded along the gray scale.

This visualization demonstrates another difference in the performance of the techniques. The co-occurrence matrix estimates all relationships as being very weak and it is difficult to differentiate strong relationships from the absence of a relationship. The half-weight index presents a much stronger set of relationships but there is evidence of relationships which have been imputed transitively. In general, HM returns a much sparser network where relationship strengths demonstrate higher contrast. This tendency towards sparsity is discussed in more detail in the Supplemental Materials S4.2.



(a) Co-Occurrence (b) Half Weight Index (c) Hub Model

Figure 3: Comparison of Results for *Dream of the Red Chamber*

The EM algorithm of HM provides very stable solutions. By selecting multiple starting points, we find that the adjacency matrix (Figure 3c) is repeatedly returned as the most likely parameter of the observed data.

The Hub Model parameter's standard deviation was estimated using the bootstrap technique. In general, the standard deviation was low. This was particularly true for $\hat{\rho}$ where the maximum standard deviation was 0.0173. Table 4 presents the standard deviation of the estimated adjacency matrix at different percentiles.

Percentile	Max	95 %	75 %	Med	25 %	5 %	Min
StDev	0.2696	0.1025	0.0374	0.0100	0.0000	0.0000	0.0000

Table 4: Percentiles of Standard Deviation in $\hat{A}$ estimated by HM for *Dream of the Red Chamber*

One of the main themes of *Dream of the Red Chamber* is the love story surrounding the protagonist Jia Baoyu (1st character in Figure 3c) and two potential fiances, the sickly Lin Daiyu (2nd character) and the "ideal" Xue Baochai (3rd character). Although Jia Baoyu shares a special bond with Lin Daiyu and has no significant emotional connection to Xue Baochai, he is ultimately tricked into marrying Xue Baochai (Hsueh-Chin, 2016). In Table 5, we present the relationships between these two girls and the other characters as estimated by the co-occurrence matrix, half weight index, and HM.

From the novel, Lin Daiyu is a sensitive girl who prefers to be alone. By contrast, Xue Baochai is a social and calculating girl. She is extremely good at interpersonal communication especially with the protagonist's mother (Lady Wang) and grandmother (Grandmother Jia) (Hsueh-Chin, 2016). These different personalities are clearly represented by the HM estimator while the other estimators do not identify this difference.

	Co-Occurrence Matrix (O)		Half Weight Index (H)		Hub ($\hat{A}$)	
	Lin Daiyu	Xue Baochai	Lin Daiyu	Xue Baochai	Lin Daiyu	Xue Baochai
Jia Baoyu	0.1728	0.1274	0.4563	0.3587	0.3113	0.2258
Lin Daiyu	1.0000	0.1109	1.0000	0.4866	1.0000	0.4072
Xue Baochai	0.1109	1.0000	0.4866	1.0000	0.4072	1.0000
Jia Yuanchun	0.0072	0.0050	0.0531	0.0449	0.0156	0.0228
Jia Tanchun	0.0439	0.0533	0.2490	0.3482	0.0915	0.4848
Shi Xiangyun	0.0590	0.0490	0.3273	0.3119	0.2194	0.2365
Miaoyu	0.0072	0.0036	0.0552	0.0337	0.0597	0
Jia Yingchun	0.0252	0.0274	0.1667	0.2141	0	0.2846
Jia Xichun	0.0187	0.0202	0.1313	0.1692	0.0102	0.2461
Wang Xifeng	0.0497	0.0526	0.1840	0.2131	0.0317	0.0697
Jia Qiaojie	0.0022	0.0022	0.0170	0.0208	0	0.0348
Li Wan	0.0367	0.0482	0.2086	0.3160	0.0580	0.3384
Qin Keqing	0.0007	0.0007	0.0052	0.0062	0	0
Grandmother Jia	0.0655	0.0648	0.2725	0.2985	0.1925	0.2820
Jia She	0.0065	0.0043	0.0449	0.0357	0	0
Jia Zheng	0.0122	0.0144	0.0701	0.0952	0.0143	0.0174
Jia Lian	0.0072	0.0036	0.0423	0.0245	0.0002	0.0073
Xiangling	0.0180	0.0252	0.1185	0.1961	0.0741	0.2344
Ping'er	0.0122	0.0209	0.0668	0.1306	0.0016	0.1643
Xue Pan	0.0043	0.0101	0.0292	0.0809	0	0
Granny Liu	0.0072	0.0050	0.0493	0.0411	0.0101	0.0113
Lady Wang	0.0490	0.0590	0.2248	0.3037	0.0224	0.2065
Aunt Xue	0.0302	0.0396	0.1806	0.2750	0.0479	0.1657
Hua Xiren	0.0403	0.0389	0.1938	0.2105	0.0283	0.1469
Qingwen	0.0166	0.0115	0.1020	0.0829	0.0155	0.0886
Yuanyang	0.0086	0.0101	0.0556	0.0763	0	0.0430
Mingyan	0.0007	0.0007	0.0053	0.0064	0	0
Zijuan	0.0317	0.0108	0.2184	0.0888	0.1775	0.0376
Concubine Zhao	0.0050	0.0058	0.0361	0.0495	0	0.0338

Table 5: Relationships of Lin Daiyu and Xue Baochai to other characters in *Dream of the Red Chamber*

7 CONCLUSION

To the best of our knowledge, Hub Models introduce an innovative approach to the task of implicit network inference. By defining a model-based generating mechanism to link the latent network to observed grouped data and applying an EM algorithm, we are able to estimate the network using this model.

Not only are the estimators easy to calculate in a reasonable amount of time, but they have a practical interpretation. The parameter ρ_i measures the probability that node v_i will form a group. A_{ij} measures the probability that a member of the population will be included in a group formed by node v_i.

The Hub Models compare favorably against existing techniques. Since the co-occurrence matrix and half weight index lack a generating mechanism to connect them to the observed grouped data, these measures often cannot detect important features of a network. By applying the Hub Model to the 18^{th} century Chinese novel *Dream of the Red Chamber*, we demonstrate that the HM is able to detect important features in the relationships between nodes in complex situations.

By the standards of statistical network analysis, the size of the adjacency matrices presented in this paper are small. An important question is how the Hub Model would perform with 10,000 or even 1,000,000 nodes. While it is computationally feasible to apply the Hub Model to populations of this size, there is a practical challenge of collecting enough observations to have sufficient statistical power.

We observe that how "small" or "large" a dataset is depends on the relationship between the number of nodes and the number of observed groups. In principle, if there are n nodes, the Hub Model must estimate n^2 parameters. If the number of observations is less than the number of nodes, multiple sets of parameters have the same likelihood and parameter estimation is unstable. In general, it is only when the number of observations exceeds the square of the number of nodes, that we have stable estimates.

This means that to estimate the Hub Model parameters of a population with hundreds of thousands of nodes, we would expect to have tens of billions of observations. Therefore, applying Hub Models directly to text or even a recommender system would be impractical.

In order to make the Hub Model useful for such large populations, some technique must be applied to reduce the number of parameters in the model. In this paper, we have placed no restrictions on the adjacency matrix. However, there are a number of restrictions which could be applied to enable us to handle populations with "small" datasets.

One major way is to make an assumption about the structure of the underlying network. For example, one might assume that the latent network is itself the result of a block model or exponential random graph model. Such an approach would create a hierarchical model for group formation.

A second way that assumptions about the structure of the underlying network could be applied is to change the dimensions of the adjacency matrix. In doing this, researchers may limit the number of nodes which can act as leaders or treat some nodes as having the same behavior.

The Hub Model can be potentially useful to model the term-document matrix in text mining. Such a matrix describes the frequency of terms that occur in a collection of documents, which is similar to the format of group data. So far many text mining techniques are based on a co-occurrence matrix created from the term-document matrix. The Hub Model may provide more meaningful estimates of the relations between terms.

Supplementary Materials

The supplemental materials contain additional details regarding the proof of Theorem 1, calculation of the estimating equations 3.8 and 3.9. Additionally, we provide data analysis for co-sponsorship of the 110^{th} Congress and a dataset of North American flora. We conclude with a discussion of identifiability, self-sparsity, and the protocol for text mining *Dream of the Red Chamber*.

Acknowledgements

This work is partially supported by NSF DMS 1513004.

Author's Statement

The views expressed in this paper are those of the authors and do not reflect the official policy or position of the US Army, the Department of Defense, or the US Government.

References

Airoldi, E. M., Blei, D. M., Fienberg, S. E., and Xing, E. P. (2008). Mixed membership stochastic blockmodels. *Journal of Machine Learning Research*, 9(Sep):1981–2014.

Anandkumar, A., Foster, D. P., Hsu, D., Kakade, S. M., and Liu, Y. (2015). A spectral algorithm from latent dirichlet allocation. *Algorithmica*, 72(1):193–214.

Bejder, L., Fletcher, D., and Brager, S. (1998). A method for testing association patterns of social animals. *Animal Behavior*, 56:719–725.

Brent, L. J. N., Lehmann, J., and Ramos-Fernandez, G. (2011). Social network analysis in the study of nonhuman primates: A historical perspective. *American Journal of Primatology*, 73:720–730.

Cairns, S. J. and Schwager, S. J. (1987). A comparison of association indices. *Animal Behavior*, 35.

Carreira-Perpinan, M. A. and Renals, S. (2000). Practical identifiability of finite mixtures of multivariate bernoulli distributions. *Neural Computation*, 12:141–152.

Choudhury, M., M., W. A., Hofman, J. M., and Watts, D. J. (2010). Inferring relevant social networks from interpersonal communication. *International World Wide Web Conference Committee*.

Colace, F., De Santo, M., Greco, L., Moscato, V., and Picariello, A. (2015). A collaborative user-centered framework for recommending items in online social networks. *Computers in Human Behavior*.

Dice, L. R. (1945). Measures of the amount of ecological association between species. *Ecology*, 26:297–302.

Frank, O. and Strauss, D. (1986). Markov graphs. *Journal of the American Statistical Asscociation*, 81:832–842.

Freeman, L. C., White, D. R., and Romney, A. K. (1989). *Research Methods in Social Network Analysis*. George Mason University Press.

Goldenberg, A., Zheng, A. X., Fienberg, S. E., and Airoldi, E. M. (2010). A survey of statistical network models. *Foundations and Trends in Machine Learning*, 2:129–233.

Handcock, M. D., Raftery, A. E., and Tantrum, J. M. (2007). Model-based clustering for social networks. *J. R. Statist. Soc. A*, 170:301–354.

Hawkes, D. (1974). *The Story of the Stone, or The Dream of the Red Chamber, Vol. 1: The Golden Days*. Penguin Classics.

Hiller, F. S. and Lieberman, G. L. (2001). *Introduction to Operations Research*. McGraw-Hill.

Hoff, P. D., Raftery, A. E., and Handcock, M. S. (2002). Latent space approaches to social network analysis. *Journal of the American Statistical Asscociation*, 97:1090–1098.

Holland, P. W., Laskey, K. B., and Leinhardt, S. (1983). Stochastic blockmodels: first steps. *Social Networks*, 5(2):109–137.

Hsueh-Chin, T. (2016). *CliffsNotes: Dream of the Red Chamber*. Houghton Mifflin Harcourt.

Jackson, M. O. (2010). *Social and economic networks*. Princeton university press.

Kolaczyk, E. D. (2009). *Statistical Analysis of Network Data: Methods and Models*. Springer.

Liu, Y., Cheng, J., Yan, C., Wu, X., and Chen, F. (2015). Research on the matthews correlation coefficients metrics of personalized recommendation algorithm evaluation. *International Journal of Hybrid Information Technology*, 8(1):163–172.

MacCarron, P. and Kenna, R. (2013). Viking sagas: Six degrees of icelandic separation-social networks from the viking era. *Significance*, pages 12–17.

McLachlan, G. J. and Krishnan, T. (2008). *The EM Algorithm and Extensions*. John Wiley and

Sons, Inc.

Newman, M. E. J. (2011). *Networks: An Introduction*. Oxford University Press.

Rabbat, M., Figueiredo, M., and Nowak, R. (2008). Network inference from co-occurrences. *IEEE Transactions on Information Technology*, 54(9):4053–4068.

Robins, G., Pattison, P., Kalish, Y., and Lusher, D. (2007). An introduction to exponential random graph (p*) models for social networks. *Social networks*, 29(2):173–191.

Teicher, H. (1961). Identifiability of mixtures. *The Annals of Mathematical Statistics*, 32(1):244–248.

Voelkl, B., Kasper, C., and Schwab, C. (2011). Network measures for dyadic interactions: Stability and reliability. *American Journal of Primatology*, 73:731–740.

Vretos, N., Nikolaidis, N., and Pitas, I. (2012). Video fingerprinting using latent dirichlet allocation and facial images. *Pattern Recognition*, 45(7):2489–2498.

Wasserman, S. and Faust, C. (1994). *Social Network Analysis: Methods and Applications*. Cambridge University Press.

Zachary, W. W. (1977). An information flow model for conflicts and fission in small groups. *Journal of Anthropological Research*, 33:452–473.

Department of Statistics

George Mason University

4400 University Drive, MS 4A7

Fairfax, VA 22030-4444

E-mail: (yzhao15@gmu.edu)

United States Army

1400 Defense Pentagon

Washington, DC 20301

E-mail: (charles.w.weko.mil@mail.mil)